Simulation of a Machine Learning Based Controller for a Fixed-Wing UAV with Distributed Sensors

A. Guerra-Langan*, S. Araujo-Estrada†, A. Richards‡, and S. Windsor§
Department of Aerospace Engineering, University of Bristol, Bristol, United Kingdom

Recent research suggests that the information obtained from arrays of sensors distributed on the wing of a fixed-wing small unmanned aerial vehicle (UAV) can provide information not available to conventional sensor suites. These arrays of sensors are capable of sensing the flow around the aircraft and it has been indicated that they could be a potential tool to improve flight control and overall flight performance. However, more work needs to be carried out to fully exploit the potential of these sensors for flight control. This work presents a 3 degrees-of-freedom longitudinal flight dynamics and control simulation model of a small fixed-wing UAV. Experimental readings of an array of pressure and strain sensors distributed across the wing were integrated in the model. This study investigated the feasibility of using machine learning to control airspeed of the UAV using the readings from the sensing array, and looked into the sensor layout and its effect on the performance of the controller. It was found that an artificial neural network was able to learn to mimic a conventional airspeed controller using only distributed sensor signals, but showed better performance for controlling changes in airspeed for a constant altitude than holding airspeed during changes in altitude. The neural network could control airspeed using either pressure or strain sensor information, but having both improved robustness to increased levels of turbulence. Results showed that some strain sensors and many pressure sensors signals were not necessary to achieve good controller performance, but that the pressure sensors near the leading edge of the wing were required. Future work will focus on replacing other elements of the flight control system with machine learning elements and investigate the use of reinforcement learning in place of supervised learning.

Nomenclature

Roman Symbols

A	Ramp signal amplitude
b	Wing span, m
C_d	Drag coefficient
$C_{l_{\delta_e}}$	Lift coefficient derivative with respect to δ_e
C_l	Lift coefficient
C_{m_0}	Pitching coefficient at zero lift
$C_{m_\alpha}, C_{m_q}, C_{m_{\delta_e}}$	Pitching moment coefficient derivative with respect to α, q and δ_e
C_m	Pitching moment coefficient
C_p	Pressure coefficient
C_t	Thrust coefficient
C_{VB}	Vertical bending moment coefficient
c	Wing mean aerodynamic chord, m
D	Drag, N
g	Gravity, m s^{-2}

*PhD Student, ana.guerra-langan@bristol.ac.uk
†Research Associate
‡Professor of Robotics and Control
§Senior Lecturer in Aerodynamics

I_{yy}	Moment of inertia about the body y axis, kg m^2
K_t	Throttle proportionality constant
L_{bw}	Transformation matrix from wind to body reference system
L_{ib}	Transformation matrix from body to inertia reference system
L_{wi}	Transformation matrix from inertia to wind reference system
L	Lift, N
M	Pitch moment, N m
m	Mass of the aircraft, kg
P	Pressure, Pa
q	Pitch rate, rad s^{-1}
S	Wing area, m^2
t_c	Ramp signal duration, s
T	Thrust, N
$[u, v, w]$	Airspeed components in body reference frame, m s^{-1}
$[u_w, v_w, w_w]$	Wind speed components in an inertial reference frame, m s^{-1}
V_a	Airspeed, m s^{-1}
V_m	Equilibrium velocity used to estimate aerodynamic coefficients, m s^{-1}
V_{a_n}	Nominal airspeed, m s^{-1}
VB	Vertical bending moment, N m
Greek Symbols	
α_0	Angle of attack at zero lift, rad
α	Angle of attack, rad
δ_e	Elevator deflection angle, rad
δ_{thr}	Throttle command
γ	Flight path angle, rad
ρ	Air density, kg m^{-3}
θ	Pitch Euler angle, rad

I. Introduction

Conventional sensing systems used in aircraft use Pitot-tubes and wind vanes to attain airflow information and aerodynamic parameters. These sensing devices are bulky, heavy and expensive and this can make them unsuitable for small unmanned aerial vehicles (UAVs) [1]. Miniaturised airflow sensor systems based on arrays of distributed sensors have been designed to overcome these limitations on small and micro UAVs, e.g. [2–5]. These arrays are inspired by the flow sensor arrays found in birds [6], bats [7] and insects[8]. Flying animals are thought to use the information from these arrays to help achieve their remarkable level of flight performance [9]. This bio-inspired technology could be very useful if applied to UAV control, improving their stability, fuel/energy consumption and overall flight performance. However, the integration of many parallel channels of airflow information into conventional flight control architectures is not straightforward. The arrays do not give direct measures of the dynamic state like a conventional sensor, so the exploitation of novel sensors on UAVs requires either the design of new controllers which can directly use the raw information coming from the sensor system, or the design of new processing algorithms that translate the sensor signals into the parameters needed for a conventional controller (e.g. [10, 11]).

Arrays of distributed sensors have been used to obtain real-time airflow information. Distributed pressure sensors in the form of flush air data sensing (FADS) system mounted on the wing of aircraft have been used in several studies in order to react pro-actively and avoid failures and degradation [12] or to estimate the air data states (static pressure, airspeed and angle of attack) using artificial neural network (ANN) architectures and look up tables [13, 14]. Pressure-based systems using a small number of sensors on 1 m span micro aerial vehicle (MAV) wing has been shown to be useful to estimate air data states [15] and predict turbulence effects [16, 17]. Other types of distributed sensor arrays mounted on wings to infer air data states include the use of hot-film flow speed sensors [2, 10], temperature

sensors [3] and distributed arrays of artificial hairs [18, 19]. ANNs have been used with all of these sensor types and have proven to be a good approach to estimate variables such as airspeed, angle of attack, angle of sideslip, lift coefficient and moment coefficient.

The information provided by distributed pressure and strain sensor arrays has been used to better understand the relationship between the dynamic states and loads of a small UAV and the airflow around it. For instance, two different UAV platforms were tested in the wind tunnel and free flight, one to measure the force (strain) and the other to measure the flow (pressure) [4]. Data showed a linear relation between α and both types of sensors up to stall conditions and an increase in the variance for higher α. In addition, the outputs of the sensors showed that both the controlled and natural gusts can be detected and that properties which are not encoded by the inertial measurement unit (IMU) can be measured or calculated. Following this, a wing model was instrumented with both distributed pressure and strain sensors and the angle of attack, airspeed, drag, lift and pitch moment were estimated using ANNs [5]. Measurements captured detachment of the flow and non-linear behaviours such as hysteresis and rate dependent effects. Both studies [4, 5] indicated that the sensor arrays provided very rich information about the airflow around the wing of the vehicle and its dynamics, and suggested that a non-conventional control architecture may be required to make full use of this information.

To explore the potential for making direct use of the signals from distributed sensors we investigated using an ANN to replace part of a conventional control system. This work investigates the viability of using a machine learning approach to control airspeed of a fixed-wing SUAV using a distributed layout of novel sensors on the wing. This study used a neural network to mimic the behaviour of a conventional controller by means of the pressure and strain sensor readings gathered in [5] and analyses the sensor layout and its effect on control performance. This is a first step in exploring the potential for replacing conventional control systems with machine learning systems that are better able to make use of the types of signals provided by distributed sensor arrays.

This paper is structures as follows. First, the sensor arrays and their integration in the model are described in Section 2. The conventional flight guidance, navigation and control (GNC) framework and its non-conventional equivalent are both described in Section 3. Results are shown and discussed in Section 4, and conclusions drawn in Section 6.

II. Wind tunnel dataset

Data from wind tunnel experiments was used to build the simulation model for this work [5]. The experiments were carried out using a semi-span wing of a WOT 4 Foam-E Mk2+ (Ripmax, Enfield, UK) radio control aircraft, instrumented with an array of 30 pressure and 4 strain sensors. Figure 1 shows the sensing arrays with the pressure sensors distributed along two sections of the wing, and the strain sensors located at the same chord distance in different span-wise sections. The instrumented wing was then tested in the wind tunnel and the sensor readings were recorded together with the aerodynamic loads given by a load cell at the root of the wing.

The dataset collected was used to estimate pressure and strain sensor readings on the wing of the aircraft and to improve the lift and drag model by adding information of the non-linear region. ANNs were used to estimate these parameters given α, V_a and q.

A. Data characteristics and correlation

The data recorded corresponds to quasi-static tests performed in the wind tunnel involving α sweeps, from from -15 to 20 degrees, repeated four times, at $V_a = [8, 10, 12, 14, 16, 18, 20]$ m s^{-1} for $q = [5, 10, 20, 30, 40, 50]$ $^\circ$ s^{-1}. Pressure and strain sensor readings were recorded together with the aerodynamic loads given by a load cell at the root of the wing. Both the pressure and strain sensor readings showed a linear relationship with α for a limited range about $\alpha = 0^\circ$ for all q values studied. For α values outside the linear region, the sensor data presented different hysteresis for different q values due to the variation in the timing of separation and reattachment of the flow.

The pressure and strain sensor readings (ΔP and VB, respectively) are a function of the angle of attack, airspeed and pitch rate as shown in Eqs. 1 and 2, where i refers to the section of the wing.

$$\Delta P_i = f(\alpha, V_a, q), \qquad i = A, B \tag{1}$$

$$VB_i = f(\alpha, V_a, q), \qquad i = A, B, C, D \tag{2}$$

Previous work [5] showed that this sensor data could be used to estimate α, airspeed, drag, lift and pitching moment, and could be beneficial to detect the onset of stall across the wing or to monitor its loading state during flight.

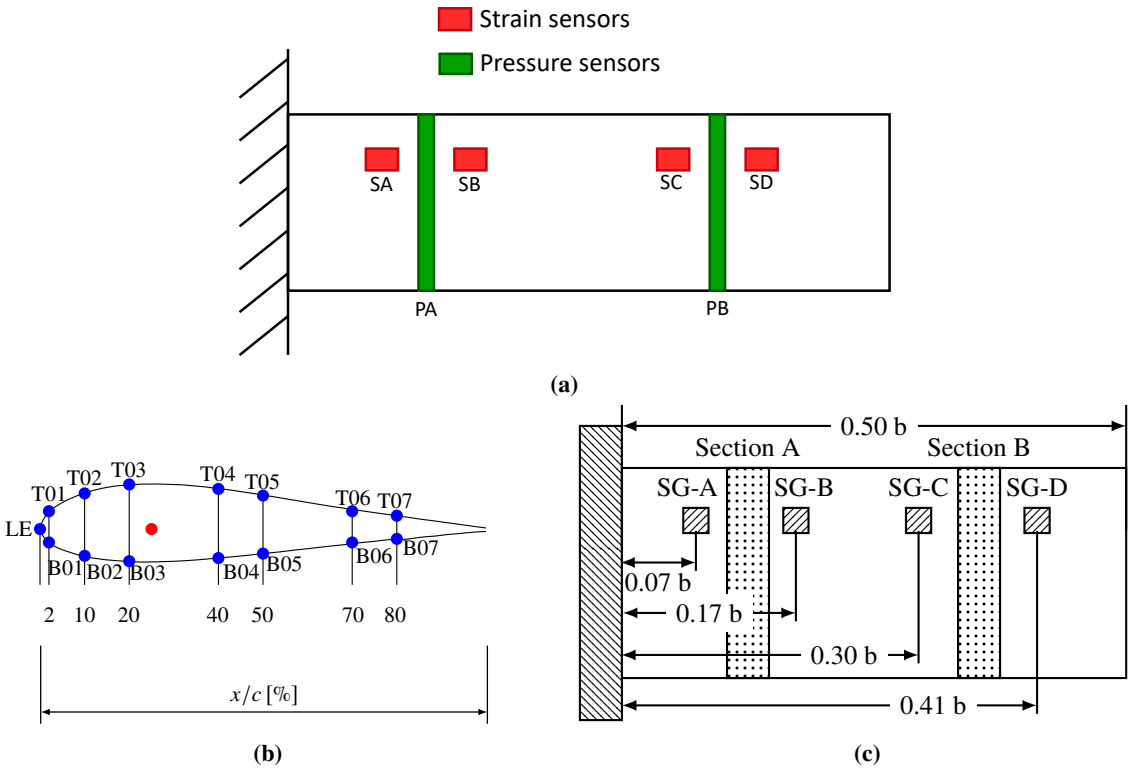


Fig. 1 Distributed sensing array: *a)* pressure and strain sensor distribution on wing, *b)* chord-wise pressure array distribution and *c)* span-wise strain array distribution.

In this work, gauge pressure readings were taken with respect to ambient pressure outside the wind tunnel. Because of this, the pressure reading at the stagnation point is equal to zero, which does not comply with Bernoulli's principle: the maximum pressure coefficient must be one at the stagnation point. Equation 3 was used to ensure that the pressure coefficients used met this condition. The strain readings were transformed into a non-dimensional vertical bending moment coefficient as per Eq. 4.

$$C_p = \frac{\Delta P_i}{\frac{1}{2}\rho V^2} + 1 \tag{3}$$

$$C_{VB} = \frac{VB_i}{\frac{1}{2}\rho V^2 Sc} \tag{4}$$

B. Pressure and strain estimation

Due to the size of the dataset and its complexity, the relationship between the state parameters and the sensor ouputs was obtained by fitting the data using ANNs. A neural network was defined for each section of the wing, two for the 30 pressure sensors and four for the 4 strain sensors used (Fig. 2). These networks were trained with the Deep Learning Toolbox in Matlab (MathWorks, MA, USA), using the scaled conjugate gradient algorithm with a maximum number of validation failures set to 50 epochs. The strain networks were defined by 5 neurons in the hidden layer and one single output node while the pressure networks were formed by 10 neurons in the hidden layer and 15 nodes in the output layer.

C. Aerodynamic load estimation

The aerodynamic loads were modelled using two sources of information. Firstly, the loads were measured in the wind tunnel with a dependency on α, V_a and q for a semi-span wing. Secondly, a mathematical model was derived from free flight data with a dependency on δ_e and δ_{thr} for the full instrumented aircraft. The experimental data obtained from

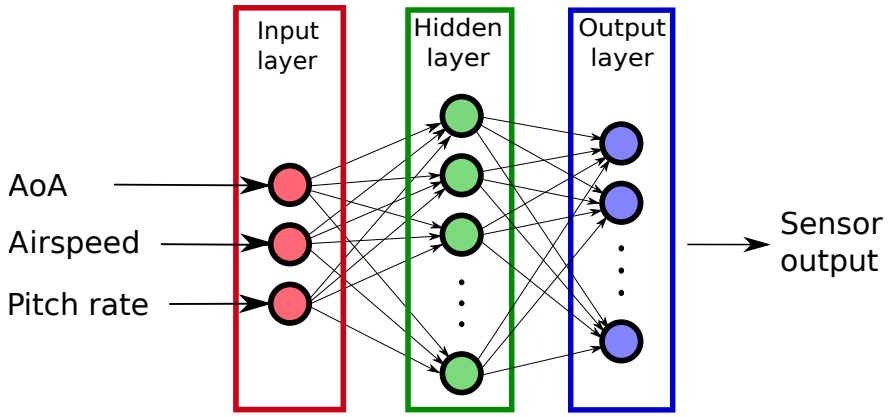


Fig. 2 Single hidden layer network used to model sensor output for given α, airspeed and pitch rate.

the wind tunnel testing was significantly richer than the outdoor flight tests, providing a wide range of V_a, α and q, including information of the non-linear region. Because of this, the lift and drag coefficients of the wind tunnel tests were integrated in the flight dynamics model. However, as the pitching moment given by the semi-span wing was not representative of the moment experienced by the full aircraft, the model derived from outdoor flight tests was used for this coefficient.

The lift and drag coefficients were correlated to α, airspeed and pitch rate using a single hidden layer neural network, following the structure in Fig. 2. It was trained using the same method as above, but with a network of 10 neurons in the hidden layer and two output neurons: C_l and C_d.

III. Framework

The framework used in this study was comprised of a non-linear flight dynamics model derived from both outdoor flight tests and wind tunnel experiments. Figure 3 below shows a block diagram of the hybrid framework used in this study. The pressure and strain sensor readings from the 1 DOF wind tunnel tests was integrated in the framework in the form of pressure and vertical bending moment coefficients as described in Section II.A.

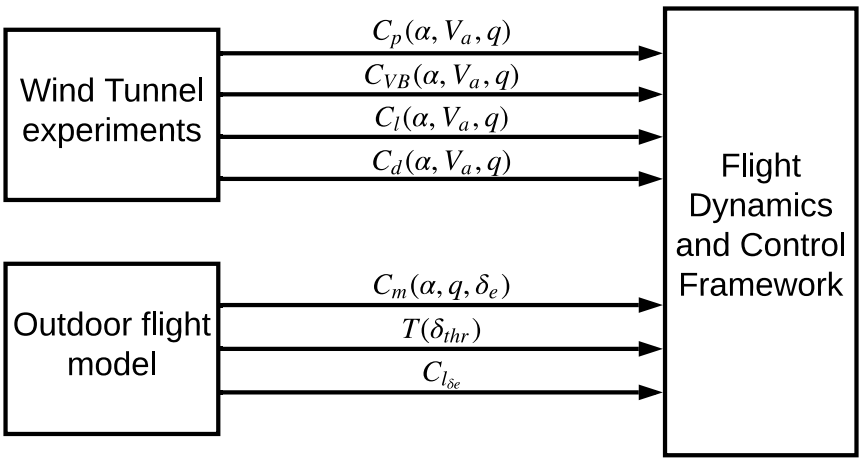


Fig. 3 Framework block diagram

The simulation framework used in this work was originally developed to study the control effort required by a powered fixed-wing UAV to exploit orographic soaring [20]. It has been reduced to 3 DOF and modified to include the pressure and strain sensor readings from [5].

Fig. 4 Ripmax WOT4 Mk2 UAV

The conventional longitudinal flight control system was used to generate an extensive data set which was then used to train, validate and test an ANN airspeed controller. The conventional airspeed controller had direct measures of airspeed, but the signals that would be produced by the distributed sensor array were collected as the conventional controller went through a defined set of training manoeuvres. The ANN-controller was then designed using a supervised learning approach to mimic the performance of the conventional control system but using the distributed pressure and strain information given by the sensor array instead of direct measurements of airspeed.

A. Conventional controller design

The 3 DOF flight guidance, navigation and control framework is described below and the corresponding block diagrams can be found in Appendix **C.**.

The flight guidance system calculated the changes in pitch angle required to follow a predefined altitude trajectory based on the SUAVs inertial position. Altitude was controlled by means of a pitch controller adjusting the elevator. The desired pitch angle was calculated following the block diagram in Figure C.2 in Appendix **C.**.

Following the work of Langelaan *et al.* [21] and Depenbusch [22], the 3 DOF flight dynamic equations used in this work are presented in Appendix **A.**. The model did not account for some of the physical limits of the aircraft or the environment. Ground effect was not taken into account. The aircraft was modelled as a point mass, with the distribution of the wind/disturbance across the wing span not being considered in the flight dynamics of the aircraft. As described in the previous section, the lift and drag coefficients used were obtained from the wind tunnel dataset.

The flight control framework was composed of a series of controllers which allowed the aircraft to maintain steady-level-flight or change its vertical position and airspeed. Block diagrams of the controllers are presented in Fig. C.1 and C.3 in Appendix **C.**. These controllers were designed and hand tuned, following [23]. The conventional airspeed controller was tuned considering changes in altitude and airspeed and at different Dryden disturbance levels to increase robustness.

B. Distributed sensor-based controller design

The aim of this work was to develop a machine learning controller that used the information from a distributed sensor array to control the aircraft in a perturbed environment. The block diagram (Fig. 5) shows the flight dynamics and control structure used. Both the conventional and the non-conventional controllers used the same flight dynamics equations of motion and guidance system, while the control was designed differently. In this case, the pressure and strain sensor readings were estimated at each time-step using α, V_a, and q and were then fed into the airspeed ANN controller together with V_{a_d}.

Training and validation datasets were obtained by running two separate batches of simulations with a conventional controller: first, maintaining altitude constant for changing airspeed and second, changing altitude for a constant airspeed. These simulations considered different acceleration and climbing rates and were run offline with the conventional controller. The recorded data was then used to train the neural network.

The training set was defined as a set of data from which the machine learning algorithm 'learns' relationships between features and targets. Simulations were run with the conventional controller controlling 'doublet' signal (Fig. 6a) demands for V_{a_d} and h_d. The time-series of the sensor readings and throttle required was then used to train the ANN.

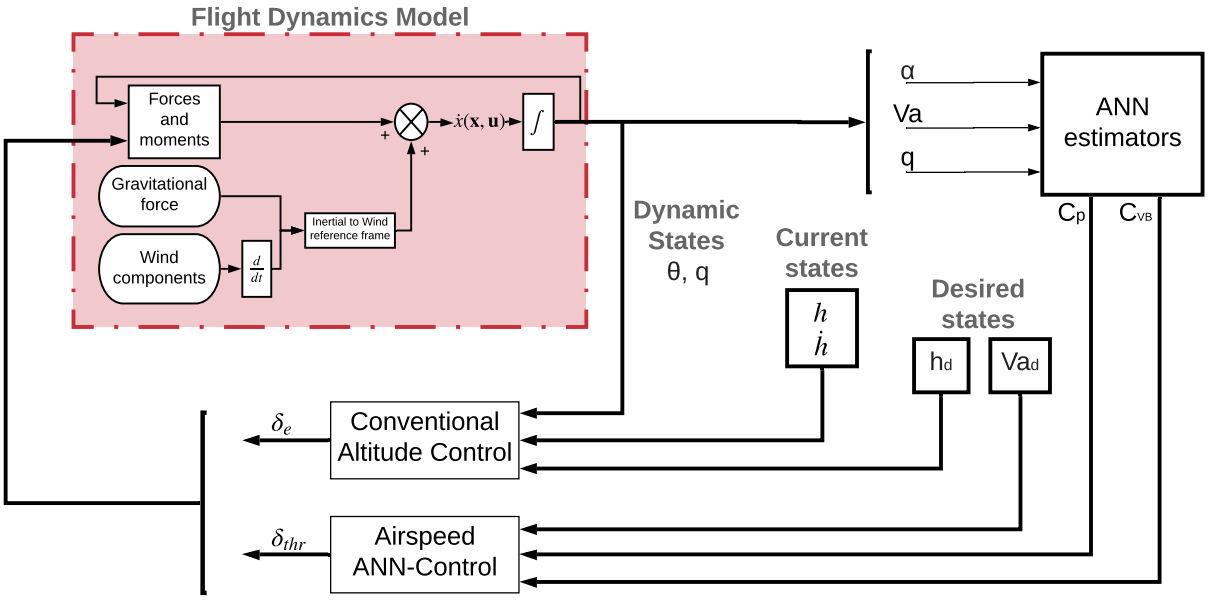


Fig. 5 Block diagram of the flight dynamics and control framework for the non-conventional controller. The inputs to the airspeed ANN-controller are the pressure and strain sensor readings, and the reference airspeed.

The validation set was a separate set of data that the machine learning algorithm was not trained on and was used to evaluate the performance of the algorithm at each iteration and to determine whether the training has reached a local optima. This dataset was also gathered through simulations with the conventional controller, and the time-series of the sensor readings and required throttle were used to evaluate the performance of the neural network by means of the root-mean-square-error (RMSE). In this case, simulations were run for '3-2-1-1' (Fig. 6b) demands in V_a and h.

Testing was done after the machine learning algorithm has been trained and validated in order to provide a final estimate of its performance. Two test cases were defined in this work in order to compare different approaches with each other: firstly, a demand of 3-2-1-1 in altitude for a constant airspeed and secondly, a 3-2-1-1 demand in airspeed for a constant altitude. Different levels of noise were also studied using the Dryden disturbance model defined in Section III.C.

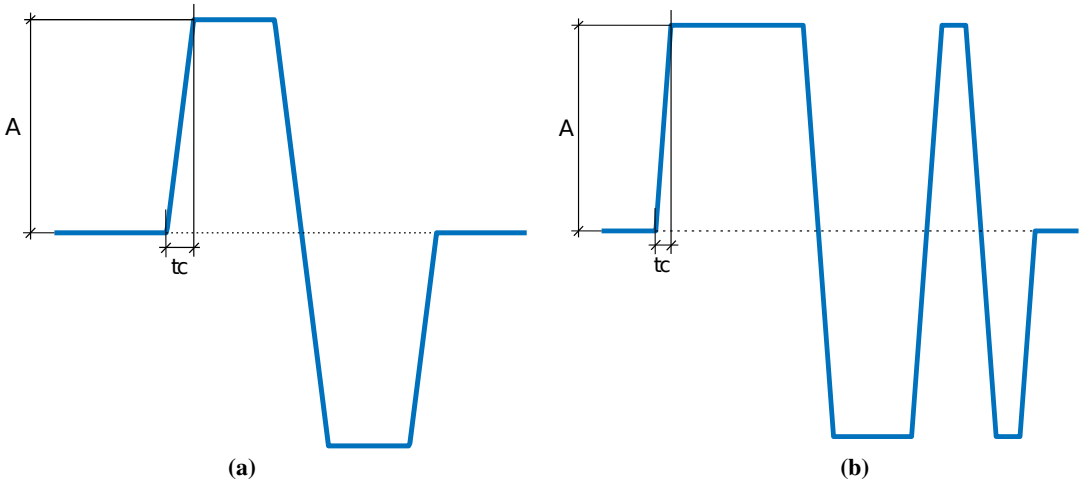


Fig. 6 Demanded signals, *a)* doublet signal used for training the ANN; *b)* '3-2-1-1' signal used for validation and testing of the ANN.

The demand signals used for training, validation and testing were based on ramp signals defined by their amplitude (A) and duration (t_c). For ramps in altitude, A represented the fraction of a 15 m change in altitude. For ramps in airspeed, A represented the fraction of a 2 m s^{-1} change in airspeed. The duration (t_c) was calculated based on the maximum climb rate and maximum acceleration of the UAV, 2.14 m s^{-1} and 0.67 m s^{-2} respectively for altitude and airspeed demands. The t_c factor represented a multiplier to the time taken to achieve the ramp (i.e. $t_c = 2$ means the aircraft was demanded to climb at half its maximum climb rate). Combinations of these values were studied for all nominal airspeeds (V_{a_n}) and Dryden levels in Table 1.

Table 1 Training, validation and testing simulations run to train the ANN-controller and check its performance.

	V_{a_n}, m s^{-1}	t_c $factor$	A $factor$	$W20$ Dryden parameter, m s^{-1}
Training set	[12 13 14 15 16 17 18]	[1 1.2 1.5 2]	[0.1 0.3 0.5 0.7 0.9 1]	0
Validation set	[13 15 17]	[1 1.25 1.75]	[0.5 1]	0
Test cases	15	1	1	[0, 2, 4, 6]

C. Dryden disturbance model

The Dryden model was used to add a continuous level of perturbation to the steady state simulations. This mathematical model represented the frequency spectrum of continuous gusts and was integrated into the flight dynamic equations of motion as an atmospheric disturbance. In this work, the MIL-F-8785C [24] specification was applied through the "Dryden Wind Turbulence Model (Continuous)" block in Simulink for low-altitude applications. The input required in each simulation was the nominal wind speed at 20 feet ($W20$) and the wind angle with respect to North. The standard deviation of the disturbance added by the Dryden model was up to 20% of $W20$ for u and up to 10% of $W20$ for w.

IV. Results and Discussion

In this work, an ANN-based airspeed controller was trained to mimic the behaviour of a conventional airspeed controller by means of distributed strain and pressure sensor readings. The sensor information gathered from wind tunnel testing was integrated in a 3 DOF longitudinal flight dynamics model in Simulink. In the course of this study, different sensor layouts were studied in order to get a better understanding of the sensors that are strictly needed for this task. Two test cases were defined, one with a 3-2-1-1 demand in altitude for constant airspeed and one with a 3-2-1-1 demand in airspeed for constant altitude. These were also run for different Dryden noise levels as per Table 1.

A. Distributed sensor-based controller

An ANN-controller that used an array of pressure and strain sensor readings on the wing to control airspeed was proposed in this work (Fig. 7). The network was trained to mimic the behaviour of a conventional airspeed controller given the sensor readings and the desired airspeed. A total of 10 ANNs with different structures were trained with three different learning algorithms. The root-mean-square-error (RMSE) for the validation set was used as selection criteria to find the ANN that best fitted the throttle required. The network chosen was trained using the Levenberg-Marquardt backpropagation function, with a maximum number of 1000 epochs and maximum validation failures set to 50. It consisted of two hidden layers of 16 neurons each with the tansig activation function and the RMSE obtained for the validation set was 2.3×10^{-2}.

Firstly, both test cases with no Dryden noise were run with the conventional controller, and the time-series data of Va_d, C_p and C_{VB} were then given to the trained neural network to test its ability to track the target throttle. Figure 8 shows the comparison between the throttle required by the conventional controller and the output of the neural network. The RMSE values of throttle required were 5.3×10^{-2} for an altitude 3-2-1-1 demand with constant airspeed, and 8.6×10^{-3} for an airspeed 3-2-1-1 demand with constant altitude.

Secondly, the response of the ANN-controller was tested by replacing the conventional airspeed controller in the simulation framework. Figure 9 shows the results obtained by the conventional and the ANN-controller in tracking the airspeed demanded for the two test cases.

Comparing both test cases, the neural network showed good performance when tracking the required throttle for a

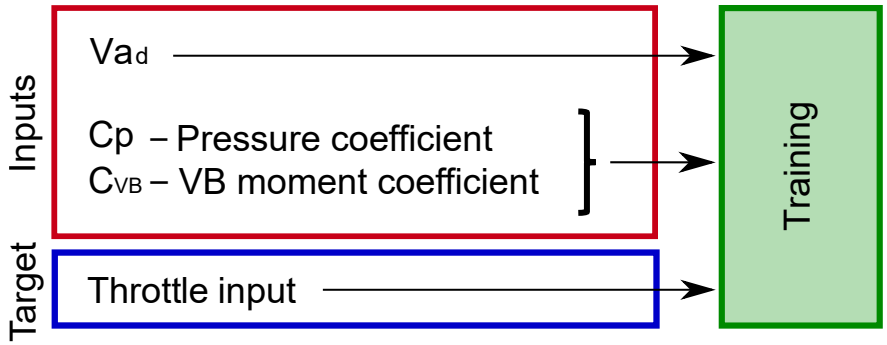
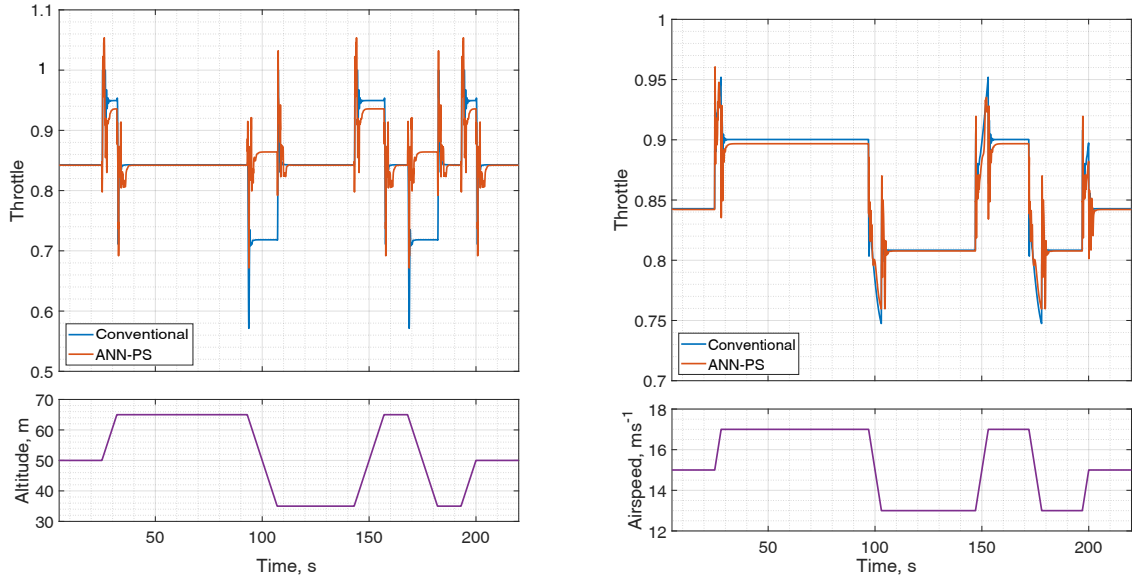


Fig. 7 Block diagram of all inputs and target fed into the initial ANN training algorithm.



(a) **(b)**

Fig. 8 Comparison between the target throttle and the ANN output for the two test cases. *a)* Throttle required for a demanded 3-2-1-1 signal in h and constant airspeed set to $15\,\mathrm{m\,s^{-1}}$; *b)* Throttle required for a demanded 3-2-1-1 signal in V_a and a constant altitude

3-2-1-1 demand in airspeed (Fig. 8), closely matching the commands of the conventional controller. In the other case, when changes in altitude were demanded the tracking error of throttle showed an increase when the desired altitude decreased; the ANN output demanded an increase of throttle instead of a decrease on its value. Because of this, the response of the ANN-controller when used in simulation showed an increase in airspeed when decreasing height (Fig. 9). The ANN-controller is accelerating the UAV instead of decelerating it to maintain constant airspeed. It is important to note that the ANN controller does not receive a direct measurement of the airspeed of the vehicle but it infers it from the pressure and strain sensor information given to it. The error in throttle and airspeed seen in Fig. 8 and 9 could be due to lack of information and may be corrected if pitching rate or altitude parameters were fed to the network.

B. Sensor Layout

Different sensor layouts were studied in order to get a better understanding of the information that is strictly necessary for an airspeed ANN-controller and the performance that could be achieved.

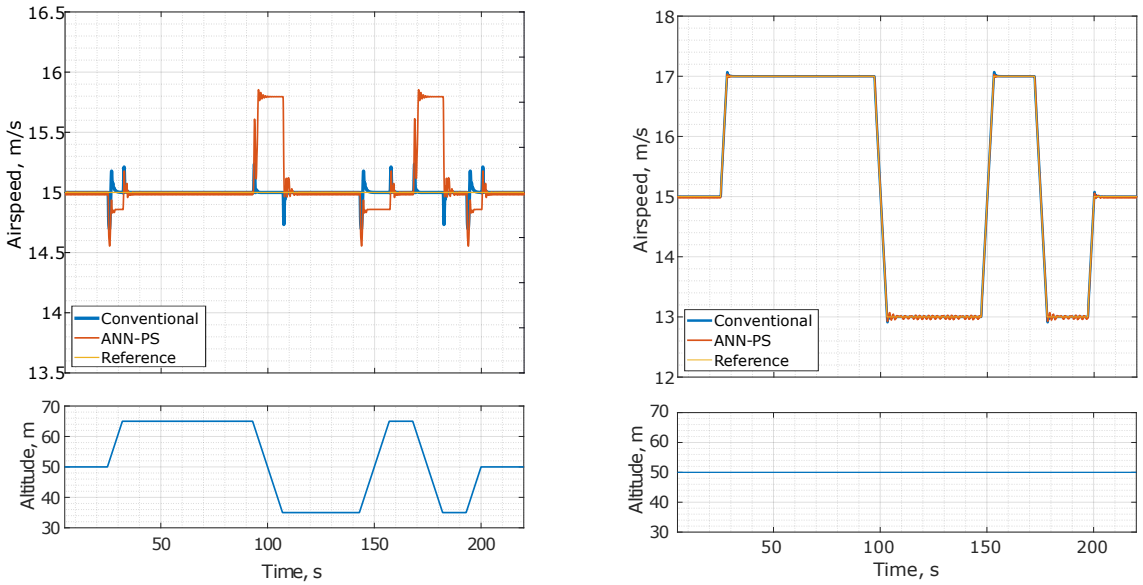


Fig. 9 Performance of conventional controller and ANN controller in simulation for two test cases. *a)* Demanded 3-2-1-1 signal in h and constant airspeed set to $15\,\mathrm{m\,s^{-1}}$; *b)* Demanded 3-2-1-1 signal in V_a and a constant altitude.

1. Strain sensor layout

Four different strain sensor layouts were studied, where the ANN was trained to control airspeed using strain information exclusively. The use of 1 to 4 sensors was investigated by keeping the sensor closer to the root (section SA of the wing) and adding the remaining sensors one at a time, sequentially towards the tip of the wing. The four ANNs were trained with the same architecture which consisted of two hidden layers of 16 neurons each with the tansig activation function. The RMSE values of airspeed for both test cases with no Dryden noise are presented in Fig. 10. The ANN-S controller (using all strain sensors) showed the best performance tracking the airspeed 3-2-1-1 demand but was considerably worse than ANN-ABC (using sensors in sections SA, SB, and SC) at maintaining constant airspeed for changes in altitude. The ANN-ABC controller performed within reasonable limits for both test cases given that the RMSE values of the conventional controller are $4 \times 10^{-2}\,\mathrm{m\,s^{-1}}$ for altitude 3-2-1-1 demand and $1.1 \times 10^{-2}\,\mathrm{m\,s^{-1}}$ for airspeed 3-2-1-1 demand. These results suggest that the strain sensor located in section SD may not be strictly necessary to control airspeed. This could be due to the vibrations of the wing tip during wind tunnel experiments, which increased the amount of outliers in the data and affected the curve fitting model integrated in the simulation framework. These outliers may be providing information that is not representative enough for the ANN to learn its relationship/effect in the airspeed control.

2. Pressure sensor layout

The study of the pressure sensor layout investigated 31 different combinations of pressure sensors. These sensor layouts were defined by removing one sensor at a time alternating from sections PA to PB, and from the top and the bottom of the wing starting from the trailing edge. The ANNs were all trained with the same architecture which consisted of one hidden layer of 16 neurons with the tansig activation function. Figure 11 shows the RMSE of airspeed for both test cases with no Dryden noise. The results show that having 30 distributed pressure sensors or just 4 next to the leading edge provides the same performance when controlling airspeed for both test cases. Figures 11c and 11d show the RMSE value of removing 0 to 26 pressure sensors, with a mean value of $9.1 \times 10^{-2} \pm 1.5 \times 10^{-2}\,\mathrm{m\,s^{-1}}$ for an altitude 3-2-1-1 demand and $5 \times 10^{-2} \pm 1.3 \times 10^{-2}\,\mathrm{m\,s^{-1}}$ for an airspeed 3-2-1-1 demand. Based on this study of pressure sensor layouts, the sensors which are essential for airspeed control are those located in the leading edge and the next closest sensor on the bottom surface of the wing for sections PA and PB (sensors B01 for sections A and B in Fig. 1b). When the signals from these sensors were removed, the airspeed error increased abruptly two orders of magnitude in the test case of altitude 3-2-1-1 demand and one order of magnitude for the airspeed 3-2-1-1 demand.

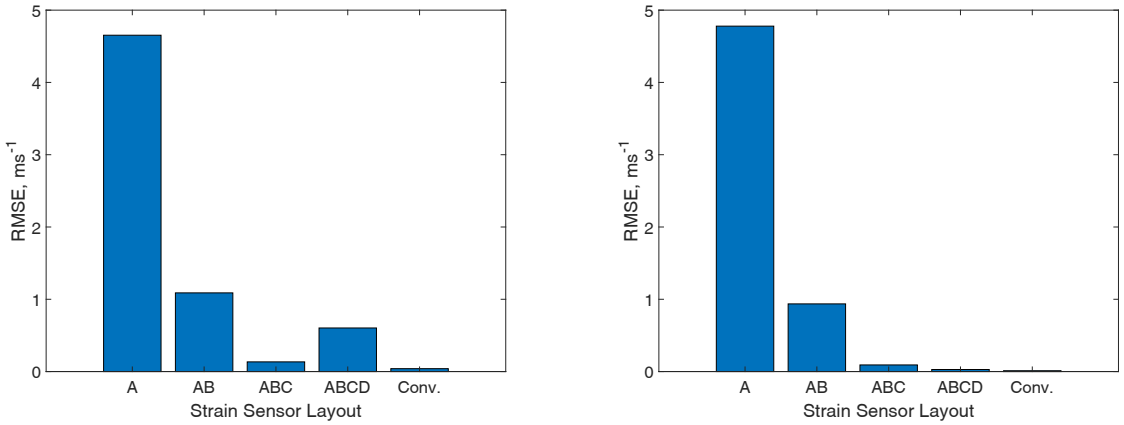


Fig. 10 RMSE values of airspeed for *a)* altitude and *b)* airspeed 3-2-1-1 demand and $W20 = 0\,\mathrm{m\,s^{-1}}$, A-D corresponds to the section of the wing in which the sensors are located and *Conv.* is the RMSE value of the conventional controller.

3. In-simulation performance comparison

Three different distributed sensor-based controllers were tested in simulation for the two test cases under three levels of Dryden noise. ANN-PS, ANN-P and ANN-S (pressure and strain, pressure only, strain only) were tested and compared with each other and with the conventional controller by means of the RMSE of airspeed. It is important to note that the ANNs were not trained with Dryden noise, hence the comparison with the conventional controller is biased. While the conventional controller was tuned accounting for different levels of noise in order to be more robust to perturbations, the ANNs need to extrapolate from the learnt information when they encounter inputs that are not within their training range. However, these results helped to get a better idea of the performance of the ANN controllers in comparison with each other and with the "ideal" RMSE value given by the conventional airspeed controller. Figure 12 shows the RMSE bars for all test cases and the four different controllers. The conventional controller outperformed the ANN-based controller in all cases and its RMSE value increased as the level of Dryden noise increased, as was to be expected. The ANN-PS performed better than the other two ANN controllers in most cases, while the ANN-S showed the worst performances, specially for the altitude 3-2-1-1 demand case where the RMSE is in the order of $1\,\mathrm{m\,s^{-1}}$. Even though the neural networks were not trained with noisy data, their performance showed a similar RMSE to the conventional controller in some cases.

4. General discussion

The results of this study demonstrate that the readings from distributed pressure and strain sensors on the wing can be used directly to control airspeed by means of an artificial neural network. The study of the sensor layout suggests that not all pressure and strain sensors are needed in order to control airspeed. However, the discarded sensors may be useful for other purposes such as aerodynamic load or dynamic states estimation, or to increase the robustness of the controller. It is also important to note that this study was realised with a longitudinal 3 DOF model of a UAV and that the discarded sensors may be more informative for lateral flight control. A more exhaustive study would need to be carried out in order to draw stronger conclusions.

In the course of this study, the validation performance during training was measured using the validation set. However, in this case, the validation performance is not representative of the controller's performance. The validation performance obtained in this study is a measurement of how well the neural network can track the target value in different cases/simulations. Nevertheless, as opposed to conventional curve fitting or regression problems, the output of the ANN-controller affects the dynamic state of the aircraft which then affects the inputs in the following time-step. Because of this, the ANN-controller may be encountering states that are out of the training range which causes the ANN to extrapolate from what it has learnt. There are two possible ways to address this in future work:

- By using simulations in the training loop for validation purposes. This would provide a better validation value

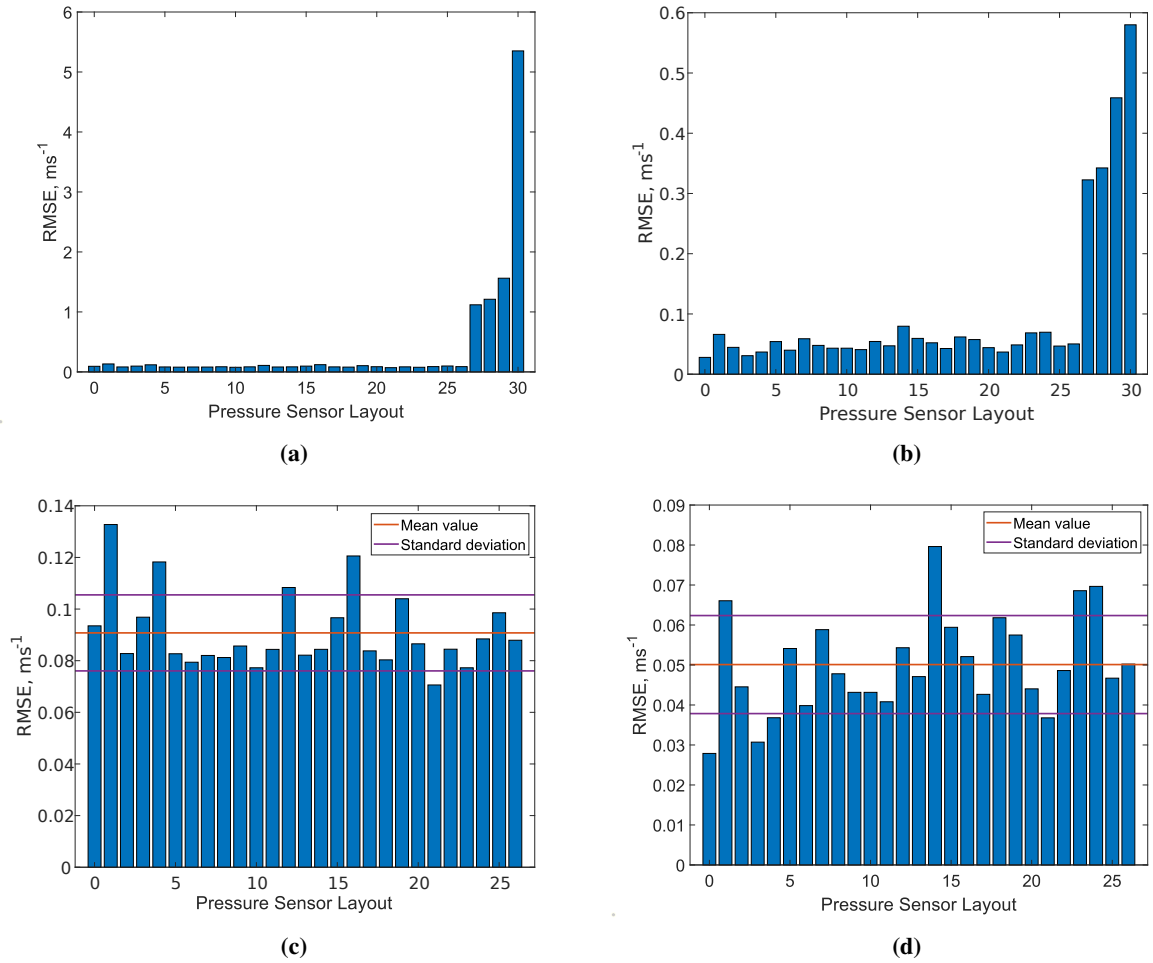


Fig. 11 RMSE values of airspeed for *a)* altitude and *b)* airspeed 3-2-1-1 demand and $W20 = 0\,\mathrm{m\,s^{-1}}$, 0-30 is the number of sensors removed from the training. Top figures show the error for 31 configurations while the bottom two show a zoom from 0-26 removed pressure sensors.

which would give a direct measurement of its control performance.

- By training in simulation using reinforcement learning. This approach would not only provide with a better measurement of control performance but could also optimise it and outperform the conventional manually-tuned controller.

V. Concluding Remarks

In this work, a 3 DOF flight dynamic model of an instrumented fixed-wing UAV was derived from wind tunnel and free-flight tests and integrated in a flight dynamics and control framework. Experimental readings of an array of pressure and strain sensors distributed across the wing were incorporated in the model. This sensor information was then used to train an ANN to mimic the behaviour of a conventional airspeed controller. Different sensor layouts were studied and the performance was measured for two test cases under different Dryden disturbance levels.

Simulations in this study suggest that the readings of distributed pressure and strain sensors can be used to control airspeed of a fixed-wing UAV. Different sensor layouts were studied and results indicated that using both pressure and strain information provides a better performance in most cases. Moreover, the sensor layout study suggested that reducing the number of pressure sensors from 30 to 4, using specific units around the leading edge of the wing, does not deteriorate the performance of the airspeed controller. In addition, using 3 instead of 4 strain sensors also improved the overall performance of the airspeed controller. However, the discarded pressure and strain sensors may be useful

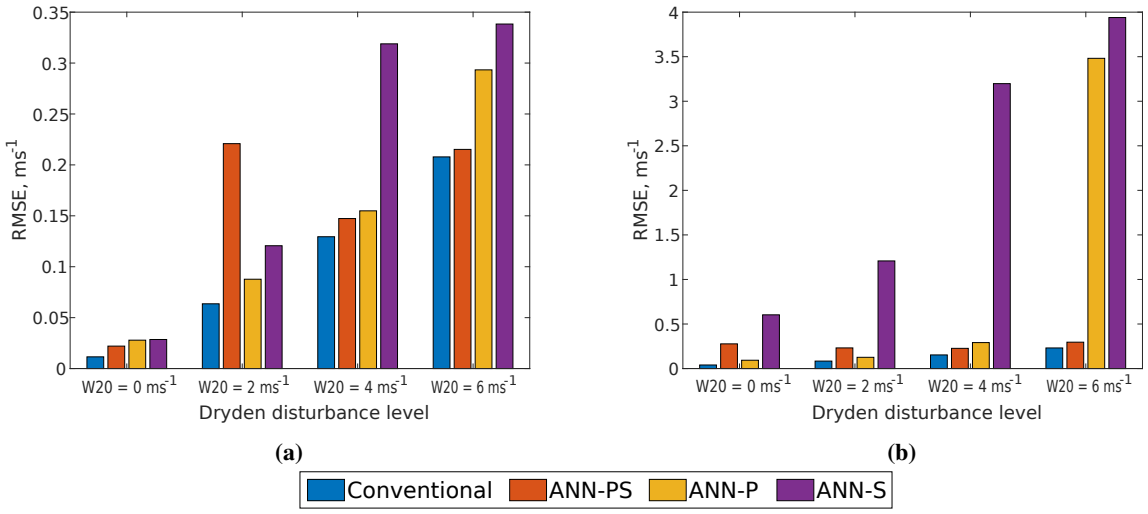


Fig. 12 RMSE values of airspeed for different controllers under different Dryden disturbance levels. ***a)*** **corresponds to the test case with 3-2-1-1 demand in altitude for constant airspeed;** ***b)*** **corresponds to the test case with a 3-2-1-1 demand in airspeed for constant altitude.**

to estimate aerodynamic loads and dynamic states or control other parameters. These sensors may also be useful to estimate or control the lateral flight dynamics states. A more extensive study needs to be carried out in order to get a better understanding of the relationship between these parameters and each individual sensor.

Future work will seek to improve the performance of the airspeed controller by means of a reinforcement learning approach. This approach could not only improve the performance of the controller but also potentially allow it to outperform the manually-tuned conventional controller presented in this work. In addition, the replacement of other conventional elements of the flight control system with machine learning elements will be explored in order to try to realise the potential flight performance improvements available from using distributed sensor arrays.

Appendix

A. Flight dynamic equations and parameters

The 3DOF flight dynamic equations used in this work are presented below, expressed in the wind reference system.

$$\dot{V_a} = -\frac{qw}{V_a} + \frac{Tcos(\alpha) - D}{m} + g_1 - \frac{du_w}{dt}d_1 - \frac{dw_w}{dt}d_3 \tag{5}$$

$$\dot{\alpha} = \frac{uq}{V_a} - \frac{Tsin(\alpha) + L}{V_a m} + \frac{g_3 - \frac{du_w}{dt}d_7 - \frac{dw_w}{dt}d_9}{V_a} \tag{6}$$

$$\dot{q} = \frac{M}{I_{yy}} \tag{7}$$

$$\dot{\theta} = q \tag{8}$$

The transformation matrix from body reference system to an inertial North-East-Down (NED) reference system and the transformation matrix from wind to body reference systems are defined in Equations 9, 10 respectively.

$$L_{ib} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \tag{9}$$

$$L_{bw} = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{10}$$

The transformation matrix from the NED inertial system to the wind reference frame:

$$L_{wi} = L_{bw}^{T} \cdot L_{ib}^{T} = \begin{bmatrix} d_1 & d_2 & d_3 \\ d_4 & d_5 & d_6 \\ d_7 & d_8 & d_9 \end{bmatrix} \tag{11}$$

Parameters d and g in the flight dynamic equations of motion (Eqs 5 - 6) are defined in Eq 11 and 12.

$$\begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} = L_{wi} \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{12}$$

Airspeed expressed in the body reference system:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = L_{bw} \cdot \begin{bmatrix} V_a \\ 0 \\ 0 \end{bmatrix} \tag{13}$$

The longitudinal aerodynamic forces and moment, and thrust are defined as:

$$\begin{bmatrix} L \\ D \end{bmatrix} = \frac{1}{2}\rho V_a^2 S \begin{bmatrix} C_l \\ C_d \end{bmatrix} \tag{14}$$

$$M = \frac{1}{2}\rho V_a^2 S c C_m \tag{15}$$

$$T = \frac{1}{2}\rho S C_t \tag{16}$$

The aerodynamic coefficients:

$$C_l = ANN(\alpha, V_a, q) + C_{l_{\delta_e}}\delta_e \tag{17}$$

$$C_d = ANN(\alpha, V_a, q) \tag{18}$$

$$C_t = K_t \delta_{thr}^2 \tag{19}$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_q}\frac{c}{2V_m}q + C_{m_{de}}\delta_e \tag{20}$$

B. WOT4 characteristics

The servo and electric motor natural frequencies and damping ratios are in Table B.1 and the simulation parameters and aerodynamic coefficients used in this work are gathered in this section in Tables B.2 and B.3, respectively.

- Pitch and control surfaces maximum and minimum angle permitted.

$$-40^\circ \leq \theta \leq 40^\circ$$
$$-15^\circ \leq \delta_e \leq 15^\circ$$

Table B.1 Servo and electric motor characteristics

Parameter	Frequency, rad s^{-1}	Damping
Elevator servo	23	0.9
Electric motor	15	0.9

Table B.2 WOT 4 simulation parameters

Physical constants		
Parameter	Value	Units
g	9.81	$m\,s^{-2}$
ρ	1.225	$kg\,m^{-3}$
Aircraft model parameters		
Parameter	Value	Units
V_m	18	$m\,s^{-1}$
S	0.3	m^2
c	0.254	m
b	1.206	m
m	1.345	kg
I_{yy}	7.8×10^{-2}	$kg\,m^2$

Table B.3 WOT 4 aerodynamic coefficients

Parameter	Value	Parameter	Value
C_{m_0}	$4.22\ \times10^{-3}$	K_t	53
C_{m_α}	$-1.01\ \times10^{-1}$	$C_{l_{\delta e}}$	$-4.24\ \times10^{-1}$
C_{m_q}	−4.84		
$C_{m_{\delta e}}$	$-3.02\ \times10^{-1}$		

C. Longitudinal flight dynamic controllers

The gains used to control the WOT 4 UAV in this study are gathered in Table C.1 and the block diagrams of the pitch angle, altitude and airspeed conventional controllers are presented in Fig. C.1-C.3.

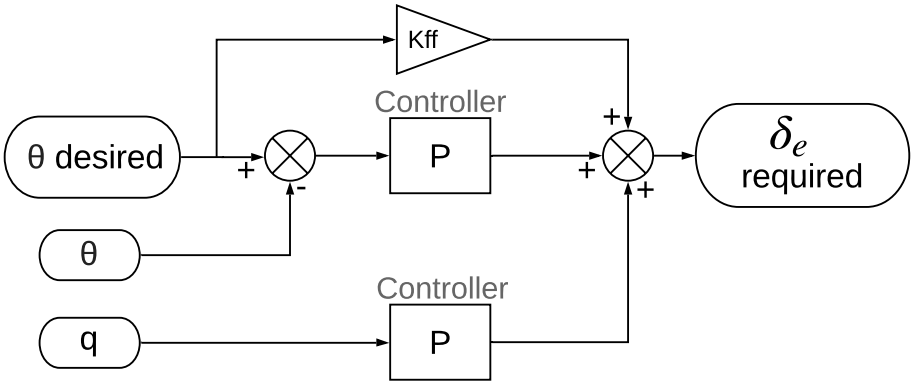


Fig. C.1 Pitch controller block diagram

Table C.1 WOT 4 PID gains

Pitch angle controller	
Parameter	Value
K_{p_θ}	−2.5
K_{p_q}	0.65
K_{ff}	−0.2
Altitude controller	
Parameter	Value
K_{p_h}	0.5
K_{i_h}	0.5
$K_{p_{\dot{h}}}$	−0.08
Velocity controller	
Parameter	Value
K_{p_V}	0.9
K_{i_V}	1

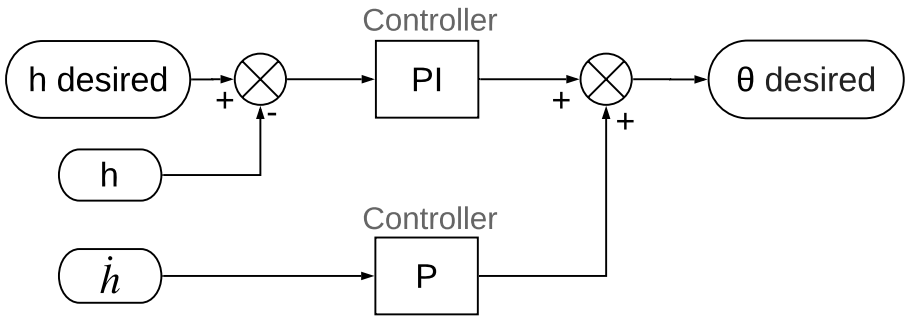


Fig. C.2 Altitude controller block diagram

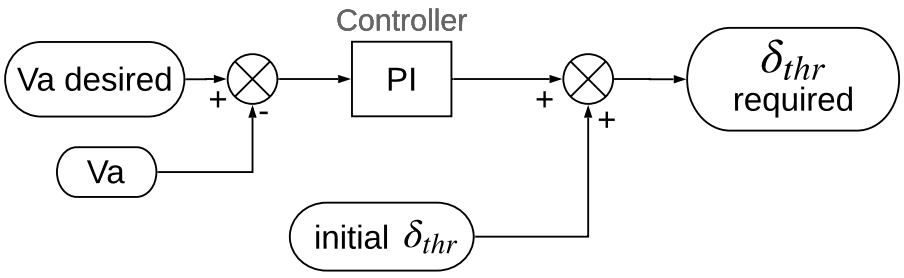


Fig. C.3 Airspeed controller block diagram

Acknowledgements

This work is funded by the EPSRC Centre for Doctoral Training in Future Autonomous and Robotic Systems (FARSCOPE) at the Bristol Robotics Laboratory.

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 679355).

References

[1] Ghommem, M., Calo, V., and Claudel, C., "Micro-cantilever flow sensor for small aircraft," *Journal of Vibration and Control*, Vol. 21, No. 10, 2015, pp. 2043–2058. doi:10.1177/1077546313505636.

[2] Fei, H., Zhu, R., Zhou, Z., and Wang, J., "Aircraft flight parameter detection based on a neural network using multiple hot-film flow speed sensors," *Smart Materials and Structures*, Vol. 16, No. 4, 2007, pp. 1239–1245. doi:10.1088/0964-1726/16/4/035.

[3] Que, R., and Zhu, R., "A Two-Dimensional Flow Sensor with Integrated Micro Thermal Sensing Elements and a Back Propagation Neural Network," *Sensors*, Vol. 14, No. 1, 2013, pp. 564–574. doi:10.3390/s140100564.

[4] Araujo-Estrada, S., Salama, F., Greatwood, C., Wood, K., Richardson, T., and Windsor, S., "Bio-inspired Distributed Strain and Airflow Sensing for Small Unmanned Air Vehicle Flight Control," *AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2017. doi:10.2514/6.2017-1487.

[5] Araujo-Estrada, S. A., and Windsor, S. P., "Aerodynamic Variables and Loads Estimation Using Bio-Inspired Distributed Sensing," *AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics, San Diego, California, 2019, pp. 1–14. doi:10.2514/6.2019-1934.

[6] Brown, R. E., and Fedde, M. R., "Airflow sensors in the avian wing," *Journal of Experimental Biology*, Vol. 179, 1993, pp. 13–30.

[7] Sterbing-D'Angelo, S., Chadha, M., Chiu, C., Falk, B., Xian, W., Barcelo, J., Zook, J. M., and Moss, C. F., "Bat wing sensors support flight control," *Proceedings of the National Academy of Sciences*, Vol. 108, No. 27, 2011, pp. 11291–11296. doi:10.1073/pnas.1018740108.

[8] Taylor, G. K., and Krapp, H. G., *Sensory Systems and Flight Stability: What do Insects Measure and Why?*, Vol. 34, 2007. doi:10.1016/S0065-2806(07)34005-8.

[9] Mohamed, A., Watkins, S., Clothier, R., Abdulrahim, M., Massey, K., and Sabatini, R., "Fixed-wing MAV attitude stability in atmospheric turbulence—Part 2: Investigating biologically-inspired sensors," *Progress in Aerospace Sciences*, Vol. 71, 2014, pp. 1–13. doi:10.1016/j.paerosci.2014.06.002.

[10] Que, R., and Zhu, R., "Aircraft Aerodynamic Parameter Detection Using Micro Hot-Film Flow Sensor Array and BP Neural Network Identification," *Sensors*, Vol. 12, No. 12, 2012, pp. 10920–10929. doi:10.3390/s120810920.

[11] Quindlen, J., and Langelaan, J., "Flush Air Data Sensing for Soaring-Capable UAVs," *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, American Institute of Aeronautics and Astronautics, Reston, Virigina, 2013, pp. 1–17. doi:10.2514/6.2013-1153.

[12] Vogel, J., and Kelkar, A., "Aircraft Control Augmentation and Health Monitoring Using Flush Air Data System Feedback," *26th AIAA Applied Aerodynamics Conference*, American Institute of Aeronautics and Astronautics, Reston, Virigina, 2008, pp. 1–15. doi:10.2514/6.2008-7505.

[13] Samy, I., Postlethwaite, I., Gu, D., and Green, J., "Neural-Network-Based Flush Air Data Sensing System Demonstrated on a Mini Air Vehicle," *Journal of Aircraft*, Vol. 47, No. 1, 2010, pp. 18–31. doi:10.2514/1.44157.

[14] Samy, I., Postlethwaite, I., and Gu, D.-W., "Unmanned air vehicle air data estimation using a matrix of pressure sensors: a comparison of neural networks and look-up tables," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 225, No. 7, 2011, pp. 807–820. doi:10.1177/0954410011399040.

[15] Marino, M., Ravi, S., and Watkins, S., "Optimum location of pressure measurements around a wing as a dynamic control input in smooth and turbulent conditions," *28th International Congress of the Aeronautical Sciences*, Vol. 2, 2012, pp. 955–962.

[16] Mohamed, A., Watkins, S., Clothier, R., and Abdulrahim, M., "Influence of Turbulence on MAV Roll Perturbations," *International Journal of Micro Air Vehicles*, Vol. 6, No. 3, 2014, pp. 175–190. doi:10.1260/1756-8293.6.3.175.

[17] Mohamed, A., Watkins, S., Fisher, A., Marino, M., Massey, K., and Clothier, R., "Bioinspired Wing-Surface Pressure Sensing for Attitude Control of Micro Air Vehicles," *Journal of Aircraft*, Vol. 52, No. 3, 2015, pp. 827–838. doi:10.2514/1.C032805.

[18] Magar, K., Reich, G., Kondash, C., Slinker, K., Pankonien, A., Baur, J., and Smyers, B., "Aerodynamic parameters from distributed heterogeneous CNT hair sensors with a feedforward neural network," *Bioinspiration & Biomimetics*, Vol. 11, No. 6, 2016, p. 066006. doi:10.1088/1748-3190/11/6/066006.

[19] Pankonien, A. M., Thapa Magar, K., Beblo, R., and Reich, G., "Gust prediction via artificial hair sensor array and neural network," *A Tribute Conference Honoring Daniel Inman*, 2017, p. 101720F. doi:10.1117/12.2257243.

[20] Guerra-Langan, A., Araujo-Estrada, S., and Windsor, S., "UAV control costs mirror bird behaviour when soaring close to buildings," *International Micro Air Vehicle Conference and Flight Competition (IMAV)*, Madrid, 2019.

[21] Langelaan, J., "Gust Energy Extraction for Mini and Micro Uninhabited Aerial Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 2, 2009, pp. 464–473. doi:10.2514/1.37735.

[22] Depenbusch, N., "Atmospheric Energy Harvesting for Small Uninhabited Aircraft by Gust Soaring," Ph.D. thesis, Pennsylvania State University, Pennsylvania, USA, 2011. doi:10.1017/CBO9781107415324.004.

[23] How, J., *16.333 Aircraft Stability and Control*, Massachusetts Institute of Technology: MIT OpenCourseWare, License: Creative Commons BY-NC-SA., 2004.

[24] *Flying Qualities of Piloted Aircraft*, Department of Defense Handbook MIL-HDBK-1797B, 2012.